What Category Theory Tells Us About Information Fusion

Steven N. Thorsen
Department of Mathematics and Statistics
Air Force Institute of Technology
Wright-Patterson AFB, OH, 45433
sthorsen@afit.edu

Mark E. Oxley
Department of Mathematics and Statistics
Air Force Institute of Technology
Wright-Patterson AFB, OH, 45433
moxley@afit.edu

Abstract - ***Category theory allows information fusion to be defined properly. Key to the definition of information fusion is the notion of the output under fusion being superior in some way to the output of simple classification systems alone. The definition therefore relies upon the construction of natural transformations suitable to the researcher. This paper discusses the meaning of such constructions.***

Keywords: Tracking, data association, estimation, neural network, resource allocation.

1 Introduction

It has been noted by several researchers (e.g., see [1, 2]) that a definition of information fusion (also referred to as data fusion) requires the final information generated to be of superior quality, in some tangible way, than the information available from the primary sources. The authors developed this idea using category theory and optimization problems in [3, 4], but did not go any further to explore the mathematical nature of tangibly measuring the required superiority. We will describe in this paper the theory necessary to explain this. The hoped for outcome of such discussion and examination is for data fusion algorithm developers to consider how they will demonstrate the superiority of the fusion algorithms to the original information sources. We must at this time restrict our discussion to classification systems (sensor -processor-classifier systems) which sense events and label them according to their inherent algorithms.

2 Information Fusion as Defined by Category Theory

2.1 Probabilistic Construction of the Event-Label Model

Let $\mathscr{C}$ be a complex of conditions [5] for a repeatable experiment, and let Ω be a set of outcomes of this experiment with $\mathrm{T} \subset \mathbb{R}$ being a bounded interval of time. Interval T sorts Ω such that we call $\mathrm{E} \subseteq \Omega \times \mathrm{T}$ an **event-state**. An event-state is then comprised of event-state elements, $e = (\omega, t) \in \mathrm{E}$, where $\omega \in \Omega$ and $t \in \mathrm{T}$. Thus e denotes a state ω at an instant of time t. Let $\Omega \times \mathrm{T}$, be the set of all event-states for an event over time interval T. Let $\mathscr{E}$ be a σ-field on $\Omega \times \mathrm{T}$, and μ be a probability measure defined on the measurable space $(\Omega \times \mathrm{T}, \mathscr{E}, \mu)$. Then the triple $(\Omega \times \mathrm{T}, \mathscr{E}, \mu)$ forms a probability space [6].

The design of a classification system involves the ability to detect (or sense) the occurrence of an event in Ω, and process the event into a label of set L. For example, design a system that detects airborne objects and classifies them friendly or unfriendly. To do this a classification system relies on several mappings, which are composed, to provide the user an answer (from the event, to the label). Since $\mathscr{E}$ is a σ-field on $\Omega \times \mathrm{T}$, then let $\mathrm{E} \in \mathscr{E}$ be any member of $\mathscr{E}$. Then a sensor, s, is defined as a mapping from E into a (raw) data set D. We denote this with the diagram

$$\mathrm{E} \xrightarrow{s} \mathrm{D}$$

so $s(e) = d \in \mathrm{D}$ for all $e \in \mathrm{E}$. The sensor is defined to produce a specific data type, so the codomain of s, $\mathrm{cod}(s) = \mathrm{D}$, where D is the set describing the data output of mapping s. A processor, p, of this system must have domain, $\mathrm{dom}(p) = \mathrm{D}$, and maps to a codomain of features, F (a refined data set), $\mathrm{cod}(p) = \mathrm{F}$. This is denoted by the diagram

$$\mathrm{D} \xrightarrow{p} \mathrm{F} \, .$$

Further, a classifier, c, of this system is a mapping such that $\mathrm{dom}(c) = \mathrm{F}$ and $\mathrm{cod}(c) = \mathrm{L}$, where L is a set of labels the user of the system finds useful. This is denoted by the diagram

$$\mathrm{F} \xrightarrow{c} \mathrm{L} \, .$$

Therefore, we can denote the entire classification system, which is diagrammed as

$$\mathrm{E} \xrightarrow{s} \mathrm{D} \xrightarrow{p} \mathrm{F} \xrightarrow{c} \mathrm{L} \, ,$$

as A, the classification system over an event-state E, where A is the composition of mappings

$$A = c \circ p \circ s.$$

Thus, A is an L-valued random variable which maps members $\mathrm{E} \in \mathscr{E}$ into the label set L and is diagrammed by

$$\mathrm{E} \xrightarrow{A} \mathrm{L} \, .$$

Consider the simple model of a multi-sensor system using two sensors in Figure 1. The sets E_i, for $i \in \{1, 2\}$, are sets of event-states. The label set L_i can be as simple as the two-class set $\{$target, non-target$\}$ or could have

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUL 2006**	2. REPORT TYPE	3. DATES COVERED **00-00-2006 to 00-00-2006**

4. TITLE AND SUBTITLE **What Category Theory Tells Us About Information Theory**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Air Force Institute of Technology,Department of Mathematics and Statistics,Wright-Patterson AFB,OH,45433**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
9th International Conference on Information Fusion, 10-13 July 2006, Florence, Italy. Sponsored by the International Society of Information Fusion (ISIF), Aerospace & Electronic Systems Society (AES), IEEE, ONR, ONR Global, Selex - Sistemi Integrati, Finmeccanica, BAE Systems, TNO, AFOSR's European Office of Aerospace Research and Development, and the NATO Undersea Research Centre.

14. ABSTRACT
see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**7**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

$$E_1 \xrightarrow{s_1} D_1 \xrightarrow{p_1} F_1 \xrightarrow{c_1} L_1$$

$$E_2 \xrightarrow{s_2} D_2 \xrightarrow{p_2} F_2 \xrightarrow{c_2} L_2$$

Figure 1: Simple Model of a Dual-Sensor System.

$$\begin{tikzcd} & D_1 \arrow[r, "p_1"] & F_1 \arrow[r, "c_1"] & L_1 \\ E \arrow[ur, "s_1"] \arrow[dr, "s_2"'] & & & \\ & D_2 \arrow[r, "p_2"] & F_2 \arrow[r, "c_2"] & L_2 \end{tikzcd}$$

Figure 2: Two Classification Systems with Overlapping Fields of View.

a more complex structure to it, such as the *types* of targets and non-targets, paired with a ranking of measure, for example [7], in order to define the battlefield more clearly for the warfighter. Now the diagram in Figure 1 represents a pair of classification systems having two sensors, two processors, and two classifiers, but can easily be extended to any finite number. Now consider two sensors not necessarily co-located. Hence they may sense different event-state sets. Figure 1 models two sensors with differing fields of view. Performing fusion along any node or edge in this graph could possibly result in an elevated level of fusion [2]–that of situation refinement or threat refinement, since we are not fusing common information about a particular event or events, but we may be fusing situations.

There are at least two other possible scenarios that Figure 1 could depict. The sensors can overlap in their field of view, either partially or fully, in which case fusing the information regarding event-states within the intersection may be useful. Thus, a fusion process may be used to increase the reliability and accuracy of the classification system, above that which is possessed by either of the sensors on its own. Let E represent that event-state set that is common to both sensors, that is, $E = E_1 \cap E_2$. Hence, there are two fundamental challenges regarding fusion. The first is how to fuse information from multiple sources regarding common event-states (or target-states, if preferred) for the purpose of knowing the event-state (presumably for the purposes of tracking, identifying, and estimating future event-states). This is commonly referred to as Level 1 fusion (or Level 0 fusion) Object Assessment. The second and much more challenging problem is to fuse information from multiple sources regarding event-states not common to all sensors, for the purpose of knowing the state of a situation (the situation-state), such as an enemy situation or threat assessment. These are the higher Levels 2 and 3, Situation Assessment and Impact Assessment. We distinguish between the two types of fusion scenarios discussed by calling them **event-state fusion** and **situation-state fusion** respectively. We will refer to mathematical models of classification systems, such as the one in Figure 2, as event-label models, so that Figure 2 represents an event-label model of a dual sensor process.

The only restriction necessary for the usefulness of this model is that a common field of view, E, be used. Consequently, D_1 and D_2 could actually be the same data set under the model, while s_1 and s_2 could be different sensors. We will refer to a finite number of families of classification systems, such as the two in Figure 2, which we wish to explore the fusion of, as a fixed classification category. For $\mathscr{E}$ considered as a category of sets, and a fixed label set L, we note that $L^{\mathscr{E}}$, is the functor category of all such classification systems, so that our fixed classification category is a subcategory of $L^{\mathscr{E}}$. Each classification system comprises a fixed branch of $L^{\mathscr{E}}$ (ie., a functor or a family of functors). Equally true is the fact that if we want to compete classification systems, we must test them over the same sample space as well. Therefore, we choose the functor category L^{E}, with a fixed L and a fixed E, to compete the classification systems over. There exists convergence theorems (e.g., see [8, 9, 10]) which allow us to treat E as if it were $\underline{\mathbf{the}}$ sample population, with the caveat that our test then is only as good as it is representative of the operational circumstances of the real-world population.

2.2 Construction of a family of classification systems

Now suppose we have a parameter $\theta \in \Theta$, which is possibly multidimensional. Then it is common that there is a family, $\{c_\theta : \theta \in \Theta\}$, of classifiers so that for each $\theta \in \Theta$, each composition,

$$c_\theta \circ p \circ s$$

describes an event-state model on fixed $E \in \mathscr{E}$, and fixed sets D, F, and L. The corresponding family

$$\mathbb{A} = \{A_\theta \mid \theta \in \Theta\},$$

where $A_\theta = c_\theta \circ p \circ s$, is a family of classification systems. Thus, Θ acts as an indexing set for defining $\mathbb{A}$. One can extend this idea to include other index sets Γ and Δ, so that the composition

$$c_\theta \circ p_\delta \circ s_\gamma,$$

where $\theta \in \Theta, \delta \in \Delta, \gamma \in \Gamma$, is a classifier, $A_{(\theta,\delta,\gamma)}$.

2.3 Defining Fusion Rules from the Event-Label Model

At this point we begin to consider categories generated by the model's sets of data. Let $\mathcal{D} = (D, \mathbf{Id}_D, \mathbf{Id}_D, \circ)$ be the discrete category generated by data set D. We use these categories to define fusion rules of classification systems.

Definition 1 (Fusion Rule of n Fixed Branches of Families of Classification Systems). Let $\mathfrak{S}_n$ be a fixed classification category with n branches. For each $i = 1, \ldots, n$, let $\mathcal{O}_i \in \mathbf{CAT}$ be a small category of data corresponding to the ith branch's source of data to be fused (this could be raw data, features, or labels). Then the product

$$\pi(n) = \prod_{i=1}^{n} \mathcal{O}_i$$

is a product category. For any particular category of data, $\mathcal{O}_0$, the exponential, $\mathcal{O}_0^{\pi(n)}$, is a category of fusion rules, each rule of which maps the products of data objects $\mathbf{Ob}(\pi(n))$ to a data object in $\mathbf{Ob}(\mathcal{O}_0)$, and maps data arrows in $\mathbf{Ar}(\pi(n))$ to arrows in $\mathbf{Ar}(\mathcal{O}_0)$. These fusion rules are functors, $\mathfrak{R}$, which make up the objects of the category. The arrows of the functor category are all the natural transformations between them. We designate $\boldsymbol{FR}_{\mathbb{O}_n}(\mathcal{O}_0)$ to be this functor category of fusion rules.

If the $\mathcal{O}_i$ are categories generated from sensor sources (*i.e.*, outputs), then we call $\mathcal{O}_0^{\pi(n)_1}$ a category of data-fusion rules and use the symbols $\mathcal{D}_0^{\pi(n)_1}$. The fusion rule branch would then be diagrammed like this:

$$E \xrightarrow{<s_1,\ldots,s_n>} \pi(n)_1 \xrightarrow{r} \mathrm{D}_0 \xrightarrow{p} \mathrm{F} \xrightarrow{c_\phi} \mathrm{L}, \tag{1}$$

where D_0 is the receiving category, r is the fusion rule, and $< s_1, \ldots, s_n >$ is the unique arrow generated by the product $\pi(n)_1$. We will not diagram any more of these, but rather note that the diagram can be written more concisely by

$$< s_1, \ldots, s_n > \circ r \circ p \circ c_\phi \tag{2}$$

If the categories are generated by processor sources, then call $\mathcal{O}_0^{\pi(n)_2}$ a category of feature-fusion rules and use the symbols $\mathcal{F}_0^{\pi(n)_2}$. This fusion rule branch is described by the composition:

$$< s_1, s_2, \ldots, s_n > \circ < p_1, p_2, \ldots, p_n > \circ r \circ c_\phi \tag{3}$$

where $\pi(n)_1$ is the product of data categories, the range of the first arrow, $\pi(n)_2$ is the product of feature categories, the range of the second arrow, r is now the fusion rule on this product of feature categories, and $< p_1, p_2, \ldots, p_n >$ is the unique arrow generated by the original processors on the product $\pi(n)_2$. Finally, if they have classifiers as sources, then call them label-fusion rules (or, alternatively, decision-fusion rules) and use the symbols $\mathcal{L}_0^{\pi(n)_3}$. This fusion rule branch is:

$$< s_1, \ldots, s_n > \circ < p_1, \ldots, p_n > \circ < c_1, \ldots, c_n > \circ r_{\boldsymbol{\phi}} \tag{4}$$

where $r_{\boldsymbol{\phi}}$ is a fusion rule for each parameter (in order to generate an appropriate family of classification systems), and $< c_1, \ldots, c_n >$ is the unique arrow generated by the original classifiers on the product $\pi(n)_3$. (We removed the parameters from the classifiers and replaced them with a single, possibly vector valued, parameter on the fusion rule).

A fusion rule could be a Boolean rule, a filter, an estimator, or an algorithm. Notice that our definition of fusion rule does not include a qualitative component; there is no necessary condition of "betterness" for a fusion rule. The result of applying a fusion rule to an existing set of fundamental branches could result in output considerably worse than existed previously. This does not affect the definition. First we define fusion rules as the key component of the fusion process. Next, we pare down the category to a subcategory which does include a qualitative component, with one suggested way of accomplishing this. We now desire to show how defining a fusor (see Definition 4) as a fusion rule with a constraint changes the Event-State model into an Event-State Fusion model. Continuing to consider the two families of classification systems in Figure 2, it is evident that a fusion rule can be designed which would apply to either the data sets, the feature sets, or the label sets (though special care needs to be taken when the actual labels are not identical in definition). Given a fusion rule $\mathfrak{R}$ for the two data sets as in Figure 2, our model becomes that of Figure 3. A new data set, processor, feature set, and classifier may become necessary as a result of the fusion rule having a different codomain than the previous systems. The label set may change also, but for now, consider a two class label set, that of

$$\mathrm{L} = \mathrm{L}_1 = \mathrm{L}_2 = \{\mathrm{Target}, \mathrm{Nontarget}\},$$

where the targets and non-targets are well-defined across classification systems (*i.e.*, each classification is identifying targets that satisfy the same definition of what a target is).

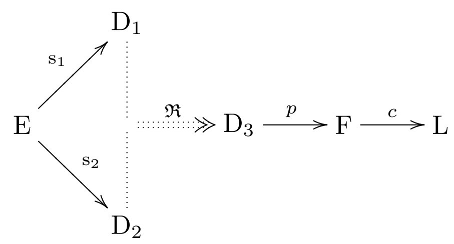

Figure 3: Fusion Rule Applied on Data Categories from Two Fixed Branches.

Now at this point we may consider, in what way is the process modeled in Figure 3 *superior* to the original processes shown in Figure 2 when $\mathrm{L} = \mathrm{L}_1 = \mathrm{L}_2$ (we will deal with the case $\mathrm{L}_1 \neq \mathrm{L}_2$ later)? One way of comparing performance in such systems is to compare the processes' receiver operating characteristics (ROC) curves, which we will show in Section 2.5.

2.4 Fusion Rules

2.4.1 Object-Fusion

There are, of course, multiple descriptions in the literature to "types" of fusion. There is *data*-fusion, *feature*-fusion, and *decision*-fusion. There is data in-feature out fusion [11] and many more. We would like to codify what should be meant by these expressions by introducing, in its most basic form, a vernacular for fusion which is intuitive, yet has its definition rooted in mathematics. We start by assuming we have a finite number of objects we wish to fuse together. What does the finite set of fusion rules look like? How can we describe in an observational way what is going on? Once the definition of fusion is established, we can move on to labeling types of fusion under certain model asumptions.

Definition 2 (Object-Fusion Category). Let $\{\mathcal{O}_i \mid i \in \{1, \ldots, m\}\}$ be a finite sequence of non-empty categories (possibly discrete). Then

$$\prod_{i=1}^{m} \mathcal{O}_i$$

defines a product category. Let

$$\pi(m) = \prod_{i=1}^{m} \mathcal{O}_i$$

for fixed $m \in \mathbb{N}$. Then for a fixed category $\mathcal{O}$, we have that

$$\mathbf{FR}_{\pi(m)}(\mathcal{O}) = \mathcal{O}^{\pi(m)}$$

is a functor category. The functor category $\mathbf{FR}_{\pi(m)}(\mathcal{O})$ is called an $\pi(m)$-Fusion category relative to $\mathcal{O}$ to denote the functors are fusing m $\mathcal{O}_i$-objects, and as necessary, their accompanying arrows into a single object and arrow in $\mathcal{O}$. When the relationship of all the $\mathcal{O}_i$ objects can be made clear, by simply calling them "objects", then we call $\mathbf{FR}_{\pi(m)}(\mathcal{O})$ the Object-Fusion category relative to $\mathcal{O}$ (regardless of the value of m).

It's important to note in our definition of fusion rules we did not put forward the notion of defining fusion *rules* in terms of performance. We will need a second mathematical definition to narrow the category of fusion rules down to a subcategory of fusion rules, which can be ordered according to their performance in some manner. First we'll consider further delineating the types of fusion rules within the Event-State model.

2.4.2 Types of Fusion Rules

We consider digraph G, as depicted in Figure 4. E is an event in the $\sigma-$field, $\mathscr{E}$. The sets D_1 and D_2 are objects of a finite collection of categories of data sets, while the sets F_1 and F_2 are objects of a finite collection of categories of feature sets. The label sets L_1 and L_2 are the objects of a finite collection of categories of label sets (and we still require that $\mathrm{L}_1 = \mathrm{L}_2$). The nodes in digraph G along which

$$\begin{tikzcd}
& \mathrm{D}_1 \arrow[r, "p_1"] & \mathrm{F}_1 \arrow[r, "c_\theta"] & \mathrm{L}_1 \\
\mathrm{E} \arrow[ur, "s_1"] \arrow[dr, "s_2"'] & & & \\
& \mathrm{D}_2 \arrow[r, "p_2"] & \mathrm{F}_2 \arrow[r, "c_\phi"] & \mathrm{L}_2
\end{tikzcd}$$

Figure 4: Digraph G.

fusion rules are generally applied are at the data, feature, and label categories. Using category theory, we can also describe that there should theoretically be nodes at the *arrows* of digraph G for fusion rules as well, though we have no non-trivial example at this time of a rule or algorithm that does this without using the pointwise outputs of the arrows. So, theoretically, we could possibly have sensor-, processor-, and classifier-fusion rules (in the sense of these being arrows and not speaking of their outputs in this manner).

2.5 Operating Characteristic Functionals

Definition 3 (Similar Families of Classification Systems). Two families of classification systems $\mathbb{A}$ and $\mathbb{B}$ are called similar if and only if they operate on the same σ-field and their output is the same well-defined label set.

Suppose we have a fixed classification category L^{E}, and let A be an object in this category. Then for L consisting of k labels, there exists a vector in $(n = k^2 - k)$-ROC space described by an n-vector v_A, where

$$v_A = (p_{2|1}(A), \ldots, p_{k|1}(A), \ldots, p_{k-1|k}(A)).$$

The proof is self-evident since E is a sample space. We call this vector the **operating characteristic** vector, and we let

$$V = \big\{ v_A \mid A \in \mathbf{Ob}(\mathrm{L}^{\mathrm{E}}) \big\} \tag{5}$$

and

$$\mathcal{V} = \big(\mathscr{P}(V), \mathbf{Ar}(V), \mathbf{Id}(V), \circ \big), \tag{6}$$

where $\mathscr{P}(V)$ is the power set of V. The category $\mathcal{V}$ is the category of operating characteristic families with undetermined non-identity arrows (we will determine them presently). Now, consider the category

$$\mathcal{C} = (\mathscr{P}(\mathbf{Ob}(\mathrm{L}^{\mathrm{E}})), \mathbf{Id}(\mathrm{L}^{\mathrm{E}}), \mathbf{Id}(\mathrm{L}^{\mathrm{E}}), \circ)$$

whose objects are sets of classification systems. Then $\mathbb{A} \in \mathbf{Ob}(\mathcal{C})$ for each family of classification systems $\mathbb{A}$. Let

$$\mathfrak{F} : \mathcal{C} \longrightarrow \mathcal{V} \tag{7}$$

be an operating characteristic functor, which maps power sets of classification systems to the set of operating characteristics associated with them. Let

$$\xi : \mathcal{V} \longrightarrow \mathcal{P} \tag{8}$$

be a functor where $\mathcal{P}$ is a poset, thought of as a category induced by a partial order, $\geq$, of its elements. Then ξ is a functor taking objects consisting of sets of operating characteristics into a value of $\mathcal{P}$. We do not need to define the rule at this point. Let $\mathbb{A}_0, \mathbb{A}_1 \in \mathcal{C}$, such that

$$\mathfrak{F}(\mathbb{A}_0) = f_{\mathbb{A}_0}$$

and

$$\mathfrak{F}(\mathbb{A}_1) = f_{\mathbb{A}_1}$$

where the outputs are families of operating characteristics. Then the diagram

$$\begin{tikzcd}
f_{\mathbb{A}_0} \arrow[r, "\xi"] \arrow[d, dotted, "g"'] & \xi(f_{\mathbb{A}_0}) = p_0 \arrow[d, "\geq"] \\
f_{\mathbb{A}_1} \arrow[r, "\xi"] & \xi(f_{\mathbb{A}_1}) = p_1
\end{tikzcd}$$

where $p_0, p_1 \in \mathcal{P}$, commutes for some unique (up to isomorphism) g. This g is an induced partial order on $\mathcal{V}$. Thus, for every pair of families of classification systems, $\mathbb{A}_0, \mathbb{A}_1 \in \mathcal{C}$, we have that the rectangle

$$\begin{tikzcd}
\mathbb{A}_0 \arrow[r, "\mathfrak{F}"] \arrow[d, dotted, "\succeq"'] & \mathfrak{F}(\mathbb{A}_0) = f_{\mathbb{A}_0} \arrow[r, "\xi"] \arrow[d, dotted, "g"'] & \xi(f_{\mathbb{A}_0}) = p_0 \arrow[d, "\geq"] \\
\mathbb{A}_1 \arrow[r, "\mathfrak{F}"] & \mathfrak{F}(\mathbb{A}_1) = f_{\mathbb{A}_1} \arrow[r, "\xi"] & \xi(f_{\mathbb{A}_1}) = p_1
\end{tikzcd} \tag{9}$$

commutes when we impose the criterion $\mathbb{A}_0 \succeq \mathbb{A}_1$ iff $(\xi \circ \mathfrak{F})(\mathbb{A}_0) \geq (\xi \circ \mathfrak{F})(\mathbb{A}_1)$, so that the functor $\xi \circ \mathfrak{F}$ is a natural transformation. It is precisely the arrows like g, which make such rectangles commute, that belong in the category $\mathcal{V}$. It is also the arrows induced from the partial order $\succeq$, which provide unique maps from one classification family to another, which will allow us to define the fusion process.

2.6 Defining Fusors

We are now in a position to define a way in which we can compete fusion rules. Suppose we have a fixed classification system such as that in Figure 2. Each branch of the system (whether fixed, or associated with a fusion rule) has a ROC manifold that can be associated with the family of classification systems, and we now have a viable means of competing each branch. If we can only choose among the two classification systems, take the one whose associated ROC functional is greater. Therefore, we can also compete these two classification systems with a new system that fuses the two data categories (or the feature or label categories for that matter) by fixing a third family of classification systems, which is based on the fusion rule, and finding the ROC functional of the event-to-label system corresponding to the fused data (features). If the fused branch's ROC functional is greater than either of the original two, then the fusion rule is a fusor. Repeating this process on a finite number of fusion rules, we discover a finite collection of fusors with associated ROC functional values. Since the subcategory of fusors is partially ordered, the best choice for a fusor is the fusor corresponding to the largest ROC functional value. Do you want to change your a priori probabilities? Simply adjust $\boldsymbol{\gamma}$ in the ROC functional's data and recalculate the BOTs for each system. Then calculate the ROC functional for each corresponding ROC and choose the largest value. The corresponding fusor is then the best fusor to select under your criteria. Therefore, given a finite collection of fusion rules, we have for fixed ROC functional data a partial ordering of fusors.

Definition 4 (Fusor over ROC Manifolds). Let $\mathbb{I} \subset \mathbb{N}$ be a finite subset of the natural numbers, with $\max \mathbb{I} = n$. Given $\{\mathbb{A}_i\}_{i \in \mathbb{I}}$ a finite collection of similar families of classification systems, let $\mathcal{O}_0^{\pi(n)}$ be the category of fusion rules associated with the product of n data sets. Let F_m be the ROC functional on the associated ROC manifolds of the families of classification systems, both original and fused, where $m = k^2 - k$, with k being the number of classes of interest in the classification problem. Let $(\boldsymbol{\gamma}, \boldsymbol{\alpha})$ be the established data for the problem. Then given that $f_{\mathbb{A}_i}$ is the ROC curve of the ith family of classification systems, and $f_{\mathfrak{R}}$ the ROC curve of the classification family $\mathbb{A}_{\mathfrak{R}}$, associated with fusion rule $\mathfrak{R} \in \mathbf{Ob}(\mathcal{O}_0^{\pi(n)})$, we say that

$$\mathbb{A}_i \succeq \mathbb{A}_j \iff F_m(f_{\mathbb{A}_i}) \geq F_m(f_{\mathbb{A}_j}) \qquad (10)$$

so that if $\mathbb{A}_{\mathfrak{R}} \succeq \mathbb{A}_i$ for all $i \in \mathbb{I}$, then $\mathfrak{R}$ is called a fusor.

There is then a category of fusors, which is a subcategory of $\mathcal{O}_0^{\pi(n)}$, and whose arrows are induced by the ROC functional, ξ, such that given objects $\mathfrak{R}$ and $\mathfrak{S}$ of this subcategory, then there exists an arrow, $\mathfrak{R} \xrightarrow{\gtrsim} \mathfrak{S}$ if and only

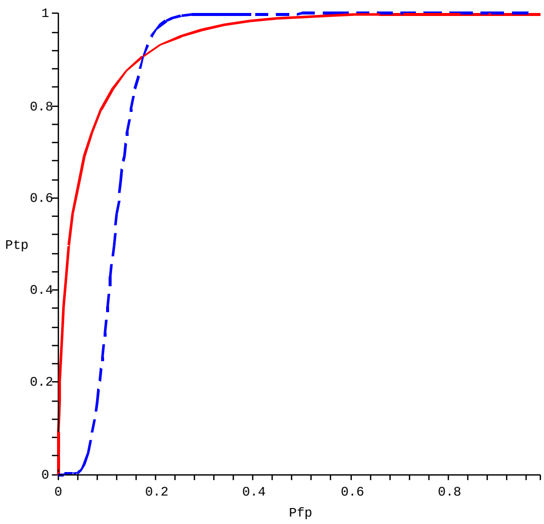

Figure 5: ROC Curves of Two Competing Classification Systems.

if $\mathbb{A}_{\mathfrak{R}} \succeq \mathbb{A}_{\mathfrak{S}}$ if and only if $p_{\mathfrak{R}} \geq p_{\mathfrak{S}}$. This can be seen in the commutativity of the rectangle constructed from Equation 9,

$$\begin{tikzcd}
\mathfrak{R} \arrow[r] \arrow[d, dotted, "\gtrsim"'] & \mathbb{A}_{\mathfrak{R}} \arrow[r, "\mathfrak{F}"] \arrow[d, dotted, "\succeq"'] & \mathfrak{F}(\mathbb{A}_{\mathfrak{R}}) = f_{\mathbb{A}_{\mathfrak{R}}} \arrow[r, "\xi"] \arrow[d, dotted, "g"'] & \xi(f_{\mathbb{A}_{\mathfrak{R}}}) = p_{\mathfrak{R}} \arrow[d, "\geq"] \\
\mathfrak{S} \arrow[r] & \mathbb{A}_{\mathfrak{S}} \arrow[r, "\mathfrak{F}"] & \mathfrak{F}(\mathbb{A}_{\mathfrak{S}}) = f_{\mathbb{A}_{\mathfrak{S}}} \arrow[r, "\xi"] & \xi(f_{\mathbb{A}_{\mathfrak{S}}}) = p_{\mathfrak{S}}
\end{tikzcd}$$

where we can see that in order for the rectangle to commute, that $\gtrsim$ must be a partial order.

We are now in a position to define the fusion processes.

Definition 5 (Fusion-Rule Process). Given a fixed classification problem defined by the category L^{E}, a fusion-rule process is an element of $\mathbf{Ob}(\mathrm{L}^{\mathrm{E}})$.

We didn't really whittle this down from the category of classification systems, because a fusion rule could be the rule "choose classification system X", which doesn't necessarily give a performance improvement. The next definition is the one of interest, since it defines the fusion with the necessary addition of a qualitative element.

Definition 6 (Fusion Process). Given a fixed classification problem defined by the category L^{E}, and a natural transformation from this category to a category defined by a poset $\mathcal{P} = (\mathrm{X}, \geq)$, let $\mathbf{FUS}_{\mathrm{L}^{\mathrm{E}}}$ be the subcategory of classification systems induced by the partial ordering. This category has as objects precisely those objects of L^{E} which have an arrow pointing to every fixed branch. We then say a fusion process is an element of $\mathbf{Ob}(\mathbf{FUS}_{\mathrm{L}^{\mathrm{E}}})$, and we can call this category the category of fusion processes.

We have now given a definition of the fusion process which contains everything necessary. As an example, suppose we start with the system

$$
\begin{tikzcd}
& \mathrm{D}_1 \arrow[r, "p_1"] & \mathrm{F}_1 \arrow[dr, "a_\theta"] & \\
\mathrm{E} \arrow[ur, "s_1"] \arrow[dr, "s_2"'] & & & \mathrm{L} \\
& \mathrm{D}_2 \arrow[r, "p_2"] & \mathrm{F}_2 \arrow[ur, "b_\phi"'] &
\end{tikzcd}
$$

with L a k-class label set. Let $A_\theta = a_\theta \circ p_1 \circ s_1$ and $B_\phi = b_\phi \circ p_2 \circ s_2$, and consider a functional F_k on the ROC curves $f_{\mathbb{A}}$ and $f_{\mathbb{B}}$ where $\mathbb{A}$ and $\mathbb{B}$ are defined as families of the respective classification systems shown (F_k being created under the assumptions and data of the researcher's choice). Then, given fusion rules $\mathfrak{S}$, such as that in Figure 6, and $\mathfrak{T}$ and a second fusion system

$$
\begin{tikzcd}
& \mathrm{D}_1 & & & & \\
\mathrm{E} \arrow[ur, "s_1"] \arrow[rr, "{\langle s_1, s_2 \rangle}"] \arrow[dr, "s_2"'] & & \mathrm{D}_1 \times \mathrm{D}_2 \arrow[ul, "q_1"'] \arrow[dl, "q_2"] \arrow[r, "\mathfrak{T}"] & \mathrm{D}_3 \arrow[r, "p_3"] & \mathrm{F}_3 \arrow[r, "d_\kappa"] & \mathrm{L} \\
& \mathrm{D}_2 & & & &
\end{tikzcd}
$$

let $f_{\mathfrak{S}}$ and $f_{\mathfrak{T}}$ refer to the corresponding ROC curves to each of the fusion rule's systems (as a possible example of ROC curves of competing fusion rules see Figure 5). This leads us to say that if we have the inequalities $F_k(f_{\mathfrak{S}}) \geq F_k(f_{\mathbb{A}})$, $F_k(f_{\mathfrak{S}}) \geq F_k(f_{\mathbb{B}})$, $F_k(f_{\mathfrak{T}}) \geq F_k(f_{\mathbb{A}})$, and $F_k(f_{\mathfrak{T}}) \geq F_k(f_{\mathbb{B}})$, then we say that $\mathfrak{S}$, $\mathfrak{T}$ are fusors. Furthermore, suppose $F_k(f_{\mathfrak{S}}) \geq F_k(f_{\mathfrak{T}})$. Then we have that $\mathfrak{S} \succeq \mathfrak{T}$. Thus, $\mathfrak{S}$ is the fusor a researcher would select under the given assumptions and data. Figure 6 is a diagram showing all branches and products (along with the associated projectors) in category theory notation.

$$
\begin{tikzcd}
& \mathrm{E} \arrow[dl, "s_1"'] \arrow[d, "{\langle s_1, s_2 \rangle}"] \arrow[dr, "s_2"] & \\
\mathrm{D}_1 \arrow[dd, "p_1"'] & \mathrm{D}_1 \times \mathrm{D}_2 \arrow[l, "q_1"'] \arrow[r, "q_2"] \arrow[d, "\mathfrak{S}"] & \mathrm{D}_2 \arrow[dd, "p_2"] \\
& \mathrm{D}_0 \arrow[d, "p"] & \\
\mathrm{F}_1 \arrow[dr, "a_\theta"'] & \mathrm{F}_0 \arrow[d, "c_\psi"] & \mathrm{F}_2 \arrow[dl, "b_\phi"] \\
& \mathrm{L} &
\end{tikzcd}
$$

Figure 6: Data Fusion of Two Classification Systems.

3 Conclusions

Fusion, by definition, requires a qualitative difference between the final output and the output of the original sources [1]. The construction of definitions for level 1 information fusion demonstrates by way of functors and natural transformations two main points.

First, a classification system should have a characteristic that can be identified as a performance characteristic of the classification system. It is essentially a functor which maps the classification system to this characteristic. If the characteristic is unique for each classification system, then one should be able to construct a second functor which maps the characteristic into a linear order. If the characteristic is not unique, we can map the characteristic into a partial order.

Second, the selected ordering (linear or partial) induces an ordering on both the characteristics and the classification systems. The usefulness of the result is dependent upon the size of the equivalence classes that are created, particularly for those values in the ordering which are likely to show up in practice. Examples of partial orders can be found with receiver operating characteristic (ROC) manifolds in [4, 3, 10]. Similarly ROC manifolds could be mapped to a linear ordering (under a ROC manifold dominance scheme), but the resulting equivalence class includes all ROC manifolds which intersect. It is thus huge and nearly worthless in practice, being desirable but only useful if your comparison happens to exhibit ROC dominance.

References

[1] Lucien Wald. Some terms of reference in data fusion. *IEEE Transactions on Geoscience and Remote Sensing*, 37(3):1190–1193, May 1999.

[2] David L. Hall and James Llinas. *Handbook of Multisensor Data Fusion*. CRC Press, Boca Raton, 2001.

[3] Steven N. Thorsen and Mark E. Oxley. A description of competing fusion systems. *Information Fusion Journal*, In press, corrected proof, 2005.

[4] Steven N. Thorsen and Mark E. Oxley. Comparing fusors within a category of fusors. In *Proceedings of the Seventh International Conference of Information Fusion*, pages 435–441, Stockholm, Sweden, June 2004. ISIF.

[5] A. N. Kolmogorov. *Foundations of the Theory of Probability*. Chelsea Publishing Company, New York, 1956.

[6] Patrick Billingsley. *Probability and Measure*. John Wiley and Sons, New York, third edition, 1995.

[7] Lei Xu and Adam Krzyák and Y. Ching Suen. Methods of combining multiple classifiers and their applications to handwriting recognition. In *IEEE Transactions on Systems, Man, and Cybernetics*, pages 418–435, May/June 1992.

[8] Stephen G. Alsing, Jr. Kenneth W. Bauer, and Mark E. Oxley. Convergence for receiver operating characteristic curves and the performance of neural networks. In *Intelligent Engineering Systems through Artificial Neural Networks*, volume 9 of *Proceedings of Artificial Neural Networks in Engineering (ANNIE) International Conference*, pages 947–952, St Louis MO, Nov 1999.

[9] Stephen G. Alsing. *The Evaluation of Competing Classifiers*. Ph.D. dissertation, Air Force Institute of Technology, Wright-Patterson AFB OH, March 2000.

[10] Steven N. Thorsen. *The Application of Category Theory and Analysis of Receiver Operating Characteristics to Information Fusion*. PhD thesis, Air Force Institute of Technology, 2950 Hobson Way, Wright-Patterson AFB, OH, 45433, Dec 2005.

[11] Belur V. Desarathy. *Decision Fusion*. IEEE Computer Society Press, 1994.